UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/824,998	04/14/2004	Filipe Fortes	MS305925.1/40062.226US01	7282

27488 7590 12/21/2007
MERCHANT & GOULD (MICROSOFT)
P.O. BOX 2903
MINNEAPOLIS, MN 55402-0903

EXAMINER
TAN, ALVIN H

ART UNIT	PAPER NUMBER
2173	

MAIL DATE	DELIVERY MODE
12/21/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/824,998	Applicant(s) FORTES ET AL.	
	Examiner Alvin H. Tan	**Art Unit** 2173	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *09 October 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-8,10-16 and 18-20* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-8,10-16 and 18-20* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *10/9/07*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Remarks

1. This Office action is responsive to the Request for Continued Examination (RCE) filed under 37 CFR §1.53(d) for the instant application on 10/9/07. Applicants have properly set forth the RCE, which has been entered into the application, and an examination on the merits follows herewith.

Claims 1, 3-8, 10-16, and 18-20 have been examined and rejected. This Office action is responsive to the amendment filed on 10/9/07, which has been entered in the above identified application.

Information Disclosure Statement

2. Publication Number US 2002/0076322 in the information disclosure statement filed 10/9/07 has not been considered because the name on the Publication does not match the name listed on the IDS. Additionally, Pub. No. US 2002/0076322 is in no way related to the application because it is directed towards a blood pump.

3. The references have been placed in the application file, but the information referred to therein has not been considered as to the merits. Applicant is advised that the date of any re-submission of any item of information contained in the information disclosure statements or the submission of any missing element(s) will be the date of submission for purposes of determining compliance with the requirements based on the

time of filing the statement, including all certification requirements for statements under 37 CFR 1.97(e). See MPEP § 609.05(a).

Claim Objections

4. The correction to claim 19 has been approved, and the objection to the claim is withdrawn.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 3-8, 10-13 and 15 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

a. Claim 3 recites the limitation "the determining operation" in *[line 1]* of the claim. It is unclear which of the determining operations in *[line 10]*, *[line 13]*, *[line 15]*, *[line 18]*, or *[line 21]* of claim 1 is being referred to.

b. Claims 3 and 15 recite the limitation "said set of constraints" in *[line 4]* of claim 3 and *[line 5]* of claim 15. It is unclear whether said set of constraints refers to the set of constraints for the parent object *[lines 7-8]* or the set of constraints of the child object *[line 16]* in claims 1 and 14.

c. Claims 3 and 15 recite the limitation "said set of properties" in *[lines 7, 9, 10]* of the claims. There is insufficient antecedent basis for this limitation in the claim.

d. Claim 15 recites the limitation "the determining operation" in *[lines 1-2]* of the claim. It is unclear which of the determining operations in *[line 10]*, *[line 13]*, *[line 15]*, *[line 20]*, or *[line 23]* of claim 14 is being referred to.

e. Claim 8 recites the limitation "the parent object" in *[line 7]* and "said parent object" in *[line 17]* of the claim. There is insufficient antecedent basis for this limitation in the claim.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

8. Claims 1, 3-8, 10-16, and 18-20 are rejected under 35 U.S.C. 102(b) as being anticipated by Joseph (U.S. Patent No. 5,873,106).

Claim 1, 3-7 (Method)

Claim 14-16, 18-20 (Machine-Readable Medium)

8-1. Regarding claims 1 and 14, Joseph teaches the claim comprising detecting a layout edit operation for a child object displayed on a video display by a computer system, by disclosing a geometry management system that handles resizing of a window including the child objects and parent containers within the window *[column 1, lines 58-67; column 2, lines 1-11]*. The geometric system supports dynamic re-layout, wherein the size of one or more objects are changed at run time due to a variation in one or more parameters that affect the geometry management of the layout *[column 3, lines 19-24]*. As described in *[column 1, lines 38-55; figures 1a, 1b]*, an object may require resizing depending on the spacing of text and graphics. Thus, a layout edit operation for a child object is detected based on parameters that affect the geometry management of the layout. Additionally, the geometry management system's visual layout mechanisms permit a user to receive direct feedback when generating a display *[column 3, lines 35-42]*. At design or layout time, child objects are placed in a parent container *[column 1, lines 65-66]*. To lay out a form, a user selects a child object, drags the child object over a selected cell, and drops the child object into the cell. In response to the user operation, the geometry management system configures the child object within the parent container such that the appropriate geometry management methods are executed *[column 5, lines 40-48]*. As described in *[column 7, lines 34-48]*, parameters control the layout of child objects. Thus, during design time, a user may edit the layout of a child object by selecting its location as well as modifying its appearance by editing parameters of its associated geometry manager.

Joseph teaches that the detecting is done via an abstraction layer implemented as an instance of an abstraction layer class, by disclosing that the geometry management system includes a set of built-in objects that operate in the overall framework of the geometry management system *[column 3, lines 25-30]* and supports add on customized geometry managers, separated from visual containers to enable mixing and matching of geometry managers with different visual containers *[column 3, lines 43-63]*, wherein calls from container to geometry managers are transparent to child objects *[column 4, lines 18-23]*. The geometry manager binds to configurable containers and implements the overall geometry management policy for that container *[column 3, lines 49-63]*. Thus, the geometry manager hides implementation details used to modify the geometry of a child object. Each geometry manager is an instance of an abstraction layer class.

Joseph teaches receiving a measure call to a parent object, wherein said measure call retrieves a set of constraints for the parent object, and an established measure parameter and an established arrange parameter of the child object, by disclosing that when a child object requires resizing, the child object calls a RequestGeometry method contained in the corresponding container object *[column 4, lines 56-58]*. A layout constraints attribute is set by a parent container object to control child objects within it *[column 4, lines 33-50]*. Based on the specified parameters of the child object, the container determines whether a given geometry is feasible *[column 4, lines 51-67]*.

Joseph teaches determining if said established measure parameter and said established arrange parameter are valid, by disclosing that a child may express preferences via a layout constraints attribute, the minimum and maximum attributes for inner and outer sizes, and via implementation of a QueryGeometry method *[column 5, lines 1-6]*. Thus, the specified parameters of the child object must meet these conditions. The specified parameters of the child objects must also conform to the constraints set by the parent container *[column 4, lines 21-23]*.

Joseph teaches when either of said established measure parameter and said established arrange parameter are set as invalid, determining if said established measure parameter is valid, and when said established measure parameter is valid, calling an arrange child helper routine to determine a final size for the child object, wherein said arrange child helper routine evaluates a set of constraints for the child object, by disclosing that even if the specified parameters of the child object are valid based on the child preferences, the child object must then go to the parent container to see whether the given geometry is feasible in order to determine a final size of the child object. Based on the constraints of the parent container, which also takes into account preferences of the child object, the child object is modified accordingly *[column 5, lines 1-6; column 6, lines 24-28]*.

Joseph teaches when said established measure parameter is invalid, calling a measure child helper routine to determine a desired size of said child object, wherein said measure child helper routine evaluates said set of constraints for the child object in respect to said set of constraints for the parent object, by disclosing that if the

dimensions of the child object are not feasible, the geometry of the child object will be modified based on the child preferences *[column 9, lines 32-44]* as well as the constraints of the parent container *[column 5, lines 1-6]*.

Joseph teaches calling said arrange child helper routine to determine said final size for the child object, by disclosing that the container/geometry manager dictates the final geometry of the child object *[column 4, lines 21-23]*.

8-2. Regarding claims 3 and 15, Joseph teaches the claim wherein the determining operation further comprises determining a container type for the parent object in which the child object is displayed, by disclosing the layout constraints attribute, which stores container/geometry manager specific attributes within a child object *[Joseph, column 9, Table 2, lines 24-31]*.

Joseph teaches retrieving said set of constraints which may include a minimum or maximum height or width or an absolute size related to the child object to be edited, by disclosing that a child may express preferences via minimum and maximum attributes for inner and outer sizes and via implementation of a QueryGeometry method *[Joseph, column 5, lines 1-6]*. These properties are used to determine the re-layout of the child object.

Joseph teaches retrieving a set of properties related to the parent object in which the child object is displayed, by disclosing that the layout constraints attribute is set by a parent container object to control child objects within it *[Joseph, column 4, lines 33-50]*.

Joseph teaches recognizing any limitations that exist within said set of properties of the parent object and said set of properties of the child object, by disclosing dimension attributes *[Joseph, column 9, lines 32-44]*. Both the parent and child parameters are taken into consideration when determining the geometry of the child object *[Joseph, column 4, lines 51-56]*.

8-3. Regarding claim 4, Joseph teaches the claim further comprising determining whether said set of constraints for the child object includes a maximum dimension, by disclosing minimum and maximum attributes for inner and outer sizes of child objects *[Joseph, column 5, lines 1-6]*.

Joseph teaches limiting adjustment of a dimension of the child object to less than or equal to the maximum dimension, by disclosing that if an object is out of a set range, the object calls set attribute calls to bring the dimensions within the permissible range *[Joseph, column 9, lines 32-44]*.

8-4. Regarding claims 5 and 16, Joseph teaches further comprising determining whether said set of constraints for the child object includes a functional relationship between the child object and the parent object, by disclosing an IsManaged attribute which determines whether the child is geometry managed by the parent of the child *[Joseph, column 9, lines 26-28]*.

Joseph teaches retrieving a reference size if said functional relationship exists and calculating a new layout parameters for the child object based on said functional

relationship and said reference size, by disclosing a Resized method that performs re-layout of child objects based on the current geometry of the container *[Joseph, column 10, lines 41-56]*. A layout constraints attribute controls the layout of the child object with respect to its container *[Joseph, column 3, lines 33-50]*. Thus the layout of child objects is based on the parameters of the container.

8-5. Regarding claim 6, Joseph teaches the claim further comprising modifying one or more properties of the child object, by disclosing that the geometric system supports dynamic re-layout, wherein the size of one or more objects are changed at run time due to a variation in one or more parameters that affect the geometry management of the layout *[Joseph, column 3, lines 19-24]*. As described in *[Joseph, column 1, lines 38-55; figures 1a, 1b]*, an object may require resizing depending on the spacing of text and graphics. Thus, the size of a child object may be edited.

8-6. Regarding claim 7, Joseph teaches the claim further comprising modifying one or more properties of the parent object, by disclosing that the geometric system supports dynamic re-layout, wherein the size of one or more objects are changed at run time due to a variation in one or more parameters that affect the geometry management of the layout *[Joseph, column 3, lines 19-24]*.

8-7. Regarding claim 18, Joseph teaches the claim further comprising determining whether a layout limitation of said child object is a proportional relationship to said

parent object and if so, maintaining said proportional relationship between a layout of said child object and said parent object, by disclosing that the geometry management policy may be to configure a container large enough to encapsulate a child object text *[Joseph, column 2, lines 4-8]*. Thus, the geometry management policy may maintain a relationship between the sizes of the parent and child object such that if the size of the child object varies, the parent object will vary in a manner that is dependent of the child object.

8-8. Regarding claim 19, Joseph teaches the claim further comprising modifying one or more properties of said child object in said measure child helper routine in the abstraction layer, by disclosing that the geometry management system specifies parameters through negotiation among the child objects, the containers, and the high level geometry manager *[Joseph, column 4, lines 51-67]*.

8-9. Regarding claim 20, Joseph teaches the claim further comprising modifying one or more properties of said child object in said arrange child helper routine in the abstraction layer consistent with one or more limitations in said parent object, by disclosing that the geometry management system specifies parameters through negotiation among the child objects, the containers, and the high level geometry manager *[Joseph, column 4, lines 51-67]*. When a child object requires resizing, the child object calls a RequestGeometry method contained in the corresponding container object. The container determines whether a given geometry is feasible and if the

parameters are acceptable, the container implements the geometry *[Joseph, column 4, lines 56-67]*.

Claims 8, 10-13 (System)

8-10. Regarding claim 8, Joseph teaches the claim comprising a processor and a memory coupled with and readable by the processor, by disclosing *[figure 7]*.

Joseph teaches detecting a layout edit operation request for a child object displayed on the video display by the computer system and sending an edit operation request to initiate layout editing of the child object, by disclosing a geometry management system that handles resizing of a window including the child objects and parent containers within the window *[column 1, lines 58-67; column 2, lines 1-11]*. The geometric system supports dynamic re-layout, wherein the size of one or more objects are changed at run time due to a variation in one or more parameters that affect the geometry management of the layout *[column 3, lines 19-24]*. As described in *[column 1, lines 38-55; figures 1a, 1b]*, an object may require resizing depending on the spacing of text and graphics. Thus, a layout edit operation for a child object is detected based on parameters that affect the geometry management of the layout. Additionally, the geometry management system's visual layout mechanisms permit a user to receive direct feedback when generating a display *[column 3, lines 35-42]*. At design or layout time, child objects are placed in a parent container *[column 1, lines 65-66]*. To layout a form, a user selects a child object, drags the child object over a selected cell, and drops the child object into the cell. In response to the user operation, the geometry

management system configures the child object within the parent container such that the appropriate geometry management methods are executed *[column 5, lines 40-48]*. As described in *[column 7, lines 34-48]*, parameters control the layout of child objects. Thus, during design time, a user may edit the layout of a child object by selecting its location as well as modifying its appearance by editing parameters of its associated geometry manager.

Joseph teaches wherein the edit operation request is sent via an application program interface, via an abstraction layer implemented as an instance of an abstraction layer class, by disclosing that the geometry management system is implemented with a plurality of objects in an object oriented program *[column 8, lines 45-52]*. The geometry management system supports add on customized geometry managers, separated from visual containers to enable mixing and matching of geometry managers with different visual containers *[column 3, lines 43-63]*, wherein calls from container to geometry managers are transparent to child objects *[column 4, lines 18-23]*. The geometry manager binds to configurable containers and implements the overall geometry management policy for that container *[column 3, lines 49-63]*. Thus, the geometry manager hides implementation details used to modify the geometry of a child object. Each geometry manager is an instance of an abstraction layer class. Since the geometry management system of Joseph is used in any environment using a windows system *[column 1, lines 11-20]*, an API would have to be used in order for the geometry management system to interface with the windows of an application.

Joseph teaches receiving a measure call to a parent object to retrieve a set of constraints for the parent object, and an established measure parameter and an established arrange parameter of the child object, by disclosing that when a child object requires resizing, the child object calls a RequestGeometry method contained in the corresponding container object *[column 4, lines 56-58]*. A layout constraints attribute is set by a parent container object to control child objects within it *[column 4, lines 33-50]*. Based on the specified parameters of the child object, the container determines whether a given geometry is feasible *[column 4, lines 51-67]*.

Joseph teaches wherein if either of said established measure parameter and said established arrange parameter are set as invalid, determining if said established measure parameter is valid, and if so, calling an arrange child helper routine to determine a final size for the child object, wherein said arrange child helper routine evaluates a set of constraints for the child object, by disclosing that even if the specified parameters of the child object are valid based on the child preferences, the child object must then go to the parent container to see whether the given geometry is feasible in order to determine a final size of the child object. Based on the constraints of the parent container, which also takes into account preferences of the child object, the child object is modified accordingly *[column 5, lines 1-6; column 6, lines 24-28]*.

Joseph teaches when said established measuring parameter is invalid, calling a measure child helper routine to determine a desired size of said child object, wherein said measure child helper routine evaluates said set of constraints for the child object in respect to said set of constraints for the parent object, by disclosing that if the

dimensions of the child object are not feasible, the geometry of the child object will be modified based on the child preferences *[column 9, lines 32-44]* as well as the constraints of the parent container *[column 5, lines 1-6]*.

Joseph teaches calling said arrange child helper routine to determine said final size for the child object, by disclosing that the container/geometry manager dictates the final geometry of the child object *[column 4, lines 21-23]*.

8-11. Regarding claim 10, Joseph teaches the claim wherein said set of constraints for the child object includes a functional relationship of size between the child object and the parent container, by disclosing that when a child object requires resizing, the child object calls a RequestGeometry method contained in the corresponding container object. The container determines whether a given geometry is feasible and if the parameters are acceptable, the container implements the geometry *[Joseph, column 4, lines 56-67]*.

8-12. Regarding claim 11 and 12, Joseph teaches the claim wherein said functional relationship is a proportional relationship between the child object and the parent container and wherein said layout editing of the child object comprises maintaining the proportional relationship between the child object and the parent container, by disclosing that the geometry management policy may be to configure a container large enough to encapsulate a child object text *[Joseph, column 2, lines 4-8]*. Thus, the geometry management policy may maintain a relationship between the sizes of the

parent and child object such that if the size of the child object varies, the parent object will vary in a manner that is dependent of the child object.

8-13. Regarding claim 13, Joseph teaches the claim wherein editing the child object comprises modifying one or more layout properties of the parent container, by disclosing that the geometry management policy may be to configure a container large enough to encapsulate a child object text *[Joseph, column 2, lines 4-8].*

Response to Arguments

9. The Examiner acknowledges the Applicant's amendments to claims 1, 3-8, 10-12, 14-16, and 18-20 and the cancellation of claim 9. Regarding independent claims 1, 8, and 14, the Applicant alleges that Joseph (U.S. Patent No. 5,873,106), as described in the previous Office action, does not explicitly teach "checking to see if either a measure parameter (the object size) or an arrange parameter (the object's position or orientation) of the child object is invalid, and when at least one is invalid and it is not the measure parameter, calling an arrange child helper routine to determine a final size for the child object", as has been amended to the claims. Contrary to Applicant's arguments, Joseph teaches that even if the specified parameters of the child object are valid based on the child preferences, the child object must then go to the parent container to see whether the given geometry is feasible in order to determine a final size of the child object. The parent container may have layout constraints related to the child object. Thus, if the child object violates the layout constraints of the parent container,

the parent container, while also taking into account the preferences of the child object, would modify the child object accordingly *[column 5, lines 1-6; column 6, lines 24-28]*.

Applicant alleges that Joseph, as described in the previous Office action, does not explicitly teach, "when the measure parameter is invalid, calling first a measure child helper routine to determine a desired size of the child object, and then calling the arrange child helper routine to determine a final size of the child object", as has been amended to the claims. Contrary to Applicant's arguments, Joseph teaches that if the dimensions of the child object are not feasible, the geometry of the child object will be modified based on the child preferences *[column 9, lines 32-44]*. Ultimately, the container/geometry manager dictates the final geometry of the child object *[column 4, lines 21-23]*.

Applicant states that dependent claims 3-7, 10-13, 15, 16, and 18-20 recite all the limitations of the independent claims, and thus, are allowable in view of the remarks set forth regarding independently amended claims 1, 8, and 14. However, as discussed above, Joseph is considered to teach claims 1, 8, and 14, and consequently, claims 3-7, 10-13, 15, 16, and 18-20 are rejected.

Conclusion

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Alvin H. Tan whose telephone number is 571-272-8595. The examiner can normally be reached on Mon-Fri 10:00-6:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Stephen Hong can be reached on 571-272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

AHT
Assistant Examiner
Art Unit 2173

TADESSE HAILU
PRIMARY EXAMINER